ИНФОРМАТИКА, ВЫЧИСЛИТЕЛЬНАЯ ТЕХНИКА И УПРАВЛЕНИЕ
INFORMATION TECHNOLOGY, COMPUTER SCIENCE, AND MANAGEMENT



УДК 004.75 10.23947/1992-5980-2017-17-4-89-96

Разработка и применение системы распределенных вычислений в решении обратных задач механики разрушений*

А. Н. Соловьев[1], П. В. Васильев[2], Л. А. Подколзина[3]**
[1,2,3] Донской государственный технический университет, г. Ростов-на-Дону, Российская Федерация

Development and application of distributed computing system in inverse problems of fracture mechanics***

A. N. Soloviev[1], P. V. Vasiliev[2], L. A. Podkolzina[3]**
[1,2,3] Don State Technical University, Rostov-on-Don, Russian Federation

Введение. При решении задач, требующих обработки большого объема данных, возникает проблема получения решения за приемлемое время. Одним из способов выполнения ограничения на временной ресурс является разделение всего объема вычислений между несколькими центрами обработки данных. Следовательно, задача разработки методов повышения быстродействия вычислительных систем и эффективности их использования является актуальной. В настоящей работе рассматривается разработка системы распределенных вычислений для обеспечения решения обратных задач в области механики разрушений. Целями работы являются проектирование и расчетно-экспериментальное обоснование системы для решения узко специализированного типа задач. В работе применяются такие программные комплексы как *ANSYS*, *COMSOL* и *FlexPDE*.

Материалы и методы Предложена методология, позволяющая использовать в качестве центров обработки обычные ПК, а не специализированные машины с предустановленным аппаратным обеспечением. Система не накладывает особых требований к аппаратной части компьютеров. Для функционирования системы необходима связь между ПК. Наличие качественной высокоскоростной сети является желательным, так как это упрощает процесс развертывания системы и увеличивает производительность вычислительного процесса. Проведен расчет конечно-элементной модели с большим набором параметров.

Результаты исследования. Разработано новое программное обеспечение для обеспечения решения обратных задач в области механики разрушений. Реализована возможность использования системы для решения более широкого спектра задач. Были учтены особенности аналогичного ПО с целью повышения отказоустойчивости и уменьшения издержек, не связанных с решением задачи. Исполняющий модуль системы проводит вычисления в многопоточном режиме, поэтому аппаратные возможности вычислительных средств используются максимально рационально. Простота организации формата хранения данных и передачи их по сети позволила достигнуть наиболее оптимального использования доступных ресурсов.

Обсуждение и заключения. Разработанная система распределенных вычислений применена при проведении конечно-

Introduction. When solving problems that require processing of a large amount of data, the problem of obtaining a solution in an acceptable time arises. One of the ways to implement a time-resource constraint is to split the entire amount of computing between several data centers. Therefore, to develop methods to improve the speed of computing systems and the effectiveness of their use is an important task. The paper deals with the development of a distributed computing system to provide solutions to inverse problems in the field of fracture mechanics. The work objectives are the design, and calculation and experimental justification of the system for solving an ad-hoc type of tasks. Such software packages as ANSYS, COMSOL, and FlexPDE are used in the work.

Materials and Methods. A methodology is proposed that allows for the use of ordinary PCs as processing centers, rather than specialized machines with preinstalled hardware. The system does not impose any special requirements on the hardware. For the system operation, the communication between PCs is necessary. The availability of the high-quality high-speed network is desirable. This simplifies the process of the system deployment, and increases the productivity of the computing process. A finite element model with a large set of parameters is calculated.

Research Results. New software is developed to provide solutions to inverse problems in the field of fracture mechanics. The opportunity to use the system for solving a wider range of tasks is realized. Special features of similar software are taken into account in order to increase resiliency and reduce costs not related to the problem solution. The system executing module performs calculations in a multithreaded mode. Therefore, the hardware capabilities of computing tools are used with maximum efficiency. Simplicity in organizing the format of data storage and transferring them over the network made it possible to achieve the most optimal utilization of the available resources.

Discussion and Conclusions. The developed system of distributed

* Работа выполнена при поддержке гранта РФФИ № 16-01-00390-а
** E-mail: solovievarc@gmail.com, lyftzigen@mail.ru, podkolzinalu@gmail.com
*** The research is done on RFFI grant no. 16-01-00390-a

элементного моделирования распространения ультразвуковой волны в стенке трубы с тонким внешним покрытием. В системе предусмотрено резервное копирование данных, что позволило сократить возможные потери расчетных данных при отказе некоторых компонентов системы.

computations is used in performing finite element modeling of the ultrasonic wave propagation in the wall of a tube with a thin outer coating. The system provides a backup of data. This made it possible to reduce the possible losses of design data when some components of the system fail.

Ключевые слова: распределенные вычисления, гетерогенные системы, конечно-элементное моделирование, неразрушающий контроль, идентификация дефектов.

Keywords: distributed computing, heterogeneous systems, finite element modeling, non-destructive testing, defect identification.

***Образец для цитирования*:** Соловьев, А. Н. Разработка и применение системы распределенных вычислений в решении обратных задач механики разрушений. А. Н. Соловьев, П. В. Васильев, Л. А. Подколзина. // Вестник Дон. гос. техн. ун-та. — 2017. — Т. 17, № 4. — С. 89–96.

For citation: A. N. Soloviev, P. V. Vasiliev, L.A. Podkolzina. Development and application of distributed computing system in inverse problems of fracture mechanics. Vestnik of DSTU, 2017, vol. 17, no.4, pp. 89–96.

Введение. В течение последних десятилетий наблюдается бурный рост многопроцессорных вычислений, включая многоядерные процессоры и создание распределенных центров обработки данных. Глобальное распространение многоядерных процессоров, мощных графических процессоров и кластеров параллельных систем в совокупности с наступившем расцветом эпохи *Big Data* и интенсивным вычислением данных открыли огромные возможности по применению и использованию параллельных и распределенных вычислений. Данные тенденции привели к тому, что параллельные и распределенные вычисления стали повсеместными в компьютерной науке, в результате чего они все чаще используются в отраслях, где возникает необходимость в обработке больших массивов данных при соблюдении жестких временных ограничений. За счет того, что параллельные и распределенные вычисления производят одновременное выполнение нескольких процессов, значительно повышается скорость работы вычислительных систем. Параллельные и распределенные вычисления основаны на базовых понятиях во многих областях вычислений, включая понимание концепций фундаментальных систем, алгоритмы, языки, системы, сети и аппаратное обеспечение, а также параллелизм и параллельное выполнение, согласованность в манипуляции с состоянием, памятью и латентностью. Взаимодействие и координация между процессами задается в моделях вычислений передачи сообщений и общей памяти и таких алгоритмических концепциях, как атомарность, консенсус и условное ожидание. Одним из самых важных условий, стоящих перед вычислительными системами, является получение решения за приемлемое время. Распределение всего объема вычислений между несколькими центрами обработки данных является эффективным способом удовлетворения условия ограничения на временной ресурс [1]. Это определяет важность задачи разработки методов повышения быстродействия и эффективности вычислительных систем.

Постоянное увеличение объемов данных повышает требования к хранению данных. Так, в докладе Международной корпорации данных указывается, что к 2020 году требования к хранению превысят 40 ЗБ [2]. Развитие технологий распределенных вычислений послужило толчком к объединению географически-распределенных по миру гетерогенных ресурсов. Благодаря появлению технических возможностей для решения масштабных задач в области науки, техники и коммерции на территориально-распределенных ресурсах, принадлежащих разным владельцам, были разработаны такие системы хранения, как распределенные файловые системы *Hadoop* [3], *Cassandra* [4] и *Hive* [5], предназначенные для решения некоторых из существующих проблем путем диспергирования (распределения) больших наборов данных по кластеру машин для облегчения последующего параллельного анализа.

Поскольку объемы данных растут вместе с необходимостью своевременного извлечения знаний из них, происходит внедрение уникальных задач обеспечения ресурсов, вычислений и управления данными. Множество организаций используют компьютерные сети, объединяющие десятки и даже сотни компьютеров. С ростом таких сетей все актуальнее становится вопрос об эффективности использования вычислительных ресурсов, которые она объединяет. Зачастую имеющиеся компьютеры не загружены полностью. Поэтому для повышения эффективности следует задействовать простаивающие ресурсы для выполнения какой-либо полезной работы — например, использовать их для решения некоторой трудоемкой задачи. Решение проблем производительности, энергоэффективности и безопасности в распределенных системах позволяет эффективно применять имеющиеся ресурсы. Развитие данного направления привело к возникновению концепции *utility computing*. Данный подход заключается в объединении независимых вычислительных ресурсов с целью получения виртуального суперкомпьютера. В последние годы популярность также набирает *Jungle Computing*, представляющая собой распределенную вычислительную парадигму, основанную на одновременном сочетании различных иерархических и распределенных вычислительных сред, которые состоят из большого количества гетерогенных ресурсов. В такой вычислительной среде ресурсы и основные вычислительные и коммуникационные инфраструктуры являются высокоиерархическими и гетерогенными. Это создает сложности при выполнении поиска надлежащих ресурсов для эффективного выполнения определенных работ в системе.

Распределенные вычислительные системы часто характеризуются множеством ресурсов, которые могут быть связаны с конкретными платформами или средами. Для максимального раскрытия потенциальных возможностей использования любых типов распределенных вычислительных ресурсов принципиально важным становится использование результативных и эффективных алгоритмов планирования, используемых менеджерами ресурсов.

В настоящее время широкое распространение получили системы, поддерживающие различные вычислительные парадигмы, такие как кластерные вычисления (*Cluster Computing*), грид-вычисления (*Grid Computing*), одноранговые вычисления (*Peer-to-Peer Computing*), а также новая концепция облачных вычислений (*Cloud Computing*), каждая из которых так или иначе содержит в себе элементы неоднородности [6–7].

Анализ *GRID*-систем. Быстрое развитие новых технологий для управления информацией и извлечения знаний, оценки и слияния, а также внедрение новых вычислительных парадигм, поддерживающих такие задачи, оказывает большое влияние на создание компьютерных систем будущего. Такие вычислительные парадигмы очень важны, особенно при выполнении семантического анализа полученной информации или предоставления распределенных или удаленных услуг с учетом предпочтений пользователей и социальных требований.

Крупномасштабные распределенные вычислительные технологии, такие как суперкомпьютеры *Cloud*, *Grid*, *Cluster* и *High Performance Computing* (*HPC*), развиваются благодаря революционному появлению многоядерных проектов (*GPU*, процессоров на одном кристалле, суперкомпьютеров на чипе и т. д.) и значительных успехов в сетях и межсоединениях (*Maassen et al*., 2011). Это приводит к усложнению интеграции разнообразных вычислительных сред, инфраструктур, платформ и технологий. Суперкомпьютеры способны решать большой спектр задач, однако за счет высокой стоимости они используются в основном для проведения научных исследований. Распространение данных, доступность оборудования, гетерогенность программного обеспечения, а также огромный объем научных проблем формируют потребность в более доступных вычислительных ресурсах, создаваемых, например, соединением кластеров в *GRID*-системы.

GRID-вычисления — это новая форма распределенных вычислений, обеспечивающая передовую вычислительную и общую модель для решения объемных и вычислительно-интенсивных задач. *GRID*-вычисления используются для предоставления различных услуг пользователям через ресурсы, которые географически распределены, динамичны и разнородны по своей природе. В *GRID*-вычислениях балансировка нагрузки играет важную роль при перераспределении пользовательских заданий при перегрузке сетевых ресурсов. В последние несколько лет разработчиками было предложено множество стратегий балансировки нагрузки, с целью улучшения времени отклика, накладных расходов на связь, пропускной способности и использования ресурсов. Несомненным плюсом данных стратегий является отсутствие выделенных ресурсов и значительных дополнительных инвестиций в оборудование. *GRID*-системы на основе персональных компьютеров (ПК) являются наиболее дешевым решением, обеспечивающим сравнительно высокую производительность в несколько петафлопс, что позволяет в большинстве ресурсоемких задач заменить ими суперкомпьютеры.

В сфере программного обеспечения существует множество различных инструментариев для организации распределенных вычислений. В качестве примеров можно привести инструментарии *Globus Toolkit, Alchemi .NET Framework, X-Com, BOINC.*

Идея применения ПК при создании *GRID*-систем основана на использовании так называемых свободных ресурсов, не участвующих в работе, выполняемой пользователем вычислительного узла. В настоящее время наиболее известной открытой программной платформой, созданной исследователями университета Беркли для реализации *GRID*-вычислений, является *BOINC*. На её основе можно создавать распределенные системы локального и глобального масштаба. Система *BOINC* состоит из программного клиента, составного сервера и специализированного ПО. Для выполнения распределенных вычислений используется клиент-серверная архитектура, где связь между сервером и клиентами осуществляется через сеть Интернет.

У всех систем подобного рода есть ряд недостатков — необходимость установки и настройки на стороне клиента специализированного программного обеспечения, сложность объединения разнородных вычислительных ресурсов (таких как *GPU*, *FPGA* и т. д.) в единую распределенную систему.

При создании собственной системы распределенных вычислений с применением платформы *.NET Framework* был использован анализ требований, предъявляемых к гетерогенной системе для распределенных вычислений, а также существующие решения в области построения *GRID*-систем.

Описание работы системы распределенных вычислений. При организации совместной работы компьютеров неизбежно возникает проблема разделения задачи на вычислительные блоки между центрами обработки. Тради-

ционно она решается созданием управляющей программы. Управляющая программа закрепляет между компьютерами вычислительные блоки задачи, организовывает их совместную работу, следит за работоспособностью компьютеров, обеспечивает сбор обработанных данных [8].

Данный подход накладывает жесткие ограничения на круг выполняемых задач из-за отсутствия мобильности в процессе перенастройки компьютеров в случае необходимости выполнения ряда задач с разными вычислительными алгоритмами. К недостаткам различных вариаций такого подхода относятся неизбежные временные потери, большие затраты на разработку программ, низкая живучесть системы из-за наличия центра управления, низкая эффективность решения вычислительных блоков задач с высокой степенью связности данных.

Разработанный алгоритм позволяет использовать в качестве центров обработки не специализированные машины с необходимым предустановленным аппаратным обеспечением, а обычные ПК, что позволяет снизить затраты на организацию вычислительных мощностей. В созданной системе нет требований к быстродействию компьютеров, а также не существует ограничений на количество ПК. К необходимым условиям относится наличие физических связей между ПК для передачи данных. Пользователь системы может применять обычную компьютерную сеть, построенную по топологии, имеющей иерархическую структуру, использующей различные сетевые технологии (например, *Ethernet*). При такой конфигурации и сетевой технологии каждый ПК может обращаться к источнику входных данных по каналу связи независимо от других ПК, передавать данные и служебную информацию от одного компьютера к другому.

Система состоит из центрального модуля, который содержит набор задач, одновременно выполняемых в рамках распределенной системы вычислений. Центральный модуль выполняет ряд функций: добавление новых задач, обработка и хранение задач, распределение задач между исполняющими модулями, отображение статистики и информации о ходе выполнения задач. Центральный модуль имеет пользовательский интерфейс, благодаря которому пользователь может в интерактивном режиме взаимодействовать и управлять им, добавлять, удалять, запускать и приостанавливать задачи. Также пользователь может как в ручном, так и в автоматическом режиме произвести сбор данных, полученных в результате решения задач. В рамки обязанностей центрального модуля входит регистрация и идентификация в сети исполняющих модулей, контроль их состояния, производительности и перераспределения задач в случае отказа какого-либо из них. Модуль поддерживает шифрование и сжатие данных, передаваемых по сети. Также, в случае необходимости, возможно установить опцию, при которой центральному модулю не будут передаваться результирующие данные. В таком случае пользователь может вручную произвести сбор данных. Центральный модуль получает информацию о производительности и аппаратном обеспечении машин, на которых установлены исполняющие модули и на основании данной информации может принять решение о распределении задач.

Исполняющий модуль представляет собой программное средство, которое устанавливается на компьютер, участвующий в процессе распределённых вычислений. Данный модуль устанавливает сетевое взаимодействие с центральным модулем и в случае готовности производить расчеты запрашивает задачи. Также он контролирует ход выполнения задачи, если это позволяет тип задачи, и производит перезапуск или отмену задания, если его не удается решить по каким-либо причинам. При первой установке соединения с центральным модулем и после авторизации исполняющий модуль получает все необходимые данные для проведения расчетов, за исключением специальных средств и программных комплексов, которые могут быть установлены вручную. Также он выполняет резервное копирование данных, полученных в результате проведения расчетов. В рамках поставленной задачи резервные копи создаются на уровне файлов. Исполняющий модуль может проводить одновременный запуск нескольких подзадач. Для обеспечения целостности данных он создает среду исполнения для каждой подзадачи и контролирует их статус выполнения.

Каждая задача имеет определенное количество подзадач. Каждая задача и подзадача уникально идентифицирована в системе во избежание коллизий. Задача представляет собой директорию с определенным набором файлов, в которых содержатся конфигурационные данные, набор параметров и дополнительная информация. Помимо этого, в директории находится программное средство, предназначенное для выполнения в фоновом режиме. Это средство производит запуск и инициализацию решаемой подзадачи. Оно может быть запушено в нескольких экземплярах.

На рис. 1 показана схема взаимодействия пользователя и модулей системы распределенных вычислений.

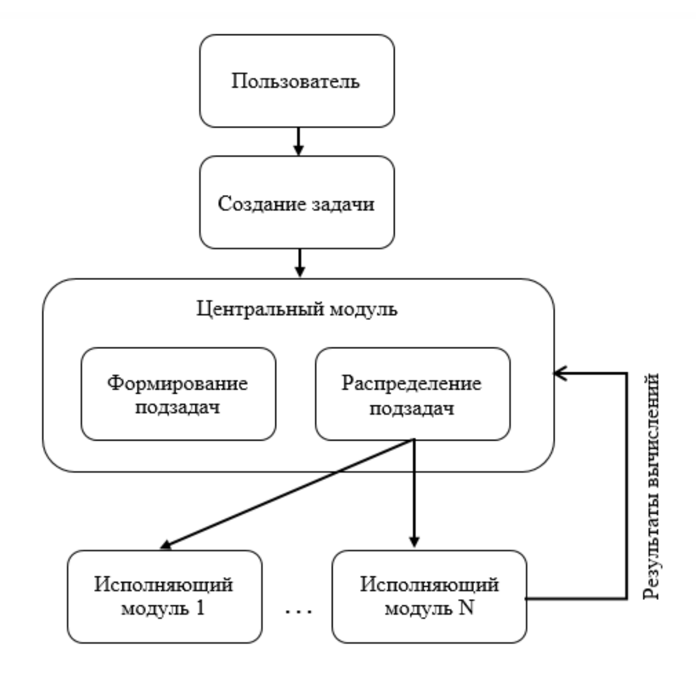


Рис. 1. Взаимодействие пользователя и модулей разработанной системы распределенных вычислений

Fig. 1. Interaction of users and modules of the developed distributed computing system

Ниже приведен алгоритм работы центрального модуля и основных его подсистем:

1. Запуск и инициализация.
2. Проверка статуса выполнения задач.
3. Проверка наличия новых задач.
4. Создание новых подзадач, если это необходимо.

Параллельно:

1. Ожидание подключения исполняемого модуля.
2. Регистрация исполняемого модуля.
3. Передача всех компонентов необходимых для решения задач.
4. Получение системной информации от исполняемого модуля.
5. Получение информации о текущих и выполненных задачах.
6. Выполнение пункта 1.

Параллельно:

1. Ожидание запроса о новой задаче от исполняемого модуля.
2. Выделение подзадачи или нескольких подзадач.
3. Выполнение пункта 1.

Параллельно:

1. Ожидание запроса на передачу данных, полученных в результате расчетов.
2. Получение данных согласно установленным параметрам.
3. Архивирование полученных данных.
4. Обновление списка выполненных подзадач и задач.
5. Оповещение исполняющих модулей о выполненных задачах.
6. Выполнение пункта 1.

Параллельно:

1. Проведение опроса подключенных исполняемых модулей.
2. Контроль состояния и сбор статистики модулей.
3. Перераспределение подзадач при необходимости.
4. Выполнение пункта 1.

Применение системы распределенных вычислений для решения обратных задач механики разрушений. Целью разработки системы распределенных вычислений является ускорение процесса решения большого набора ресурсоемких задач конечно-элементного моделирования и рациональное использование доступных вычислительных ресурсов. Неразрушающий контроль является неотъемлемой частью технического диагностирования и используется для оценки технического состояния узлов промышленного оборудования без применения разрушающих операций. Среди методов неразрушающего контроля ведущее положение занимает ультразвуковой контроль. Он основан на

способности звуковых волн отражаться от границы раздела двух упругих сред, обладающих разными акустическими свойствами [9–11].

Симуляция распространения ультразвуковой волны в твердых телах может быть проведена методом конечных элементов (МКЭ). В качестве инструмента исследования был выбран программный комплекс *COMSOL*. С помощью этого программного комплекса решались задачи моделирования распространения ультразвуковой волны в трубе, усиленной тонкой кольцевой накладкой.

Рассмотрим задачу идентификации дефекта с последующей его локализацией. Эта проблема относится к области неразрушающего контроля свойств и параметров объекта в обратных задачах механики деформируемого твердого тела. Возникает необходимость проведения большого количества ресурсоемких задач в ходе решения задачи идентификации дефектов. Необходимо провести расчет конечно-элементной модели с большим набором параметров. На рис. 2 проиллюстрировано распространение ультразвуковой волны при различных параметрах дефекта.

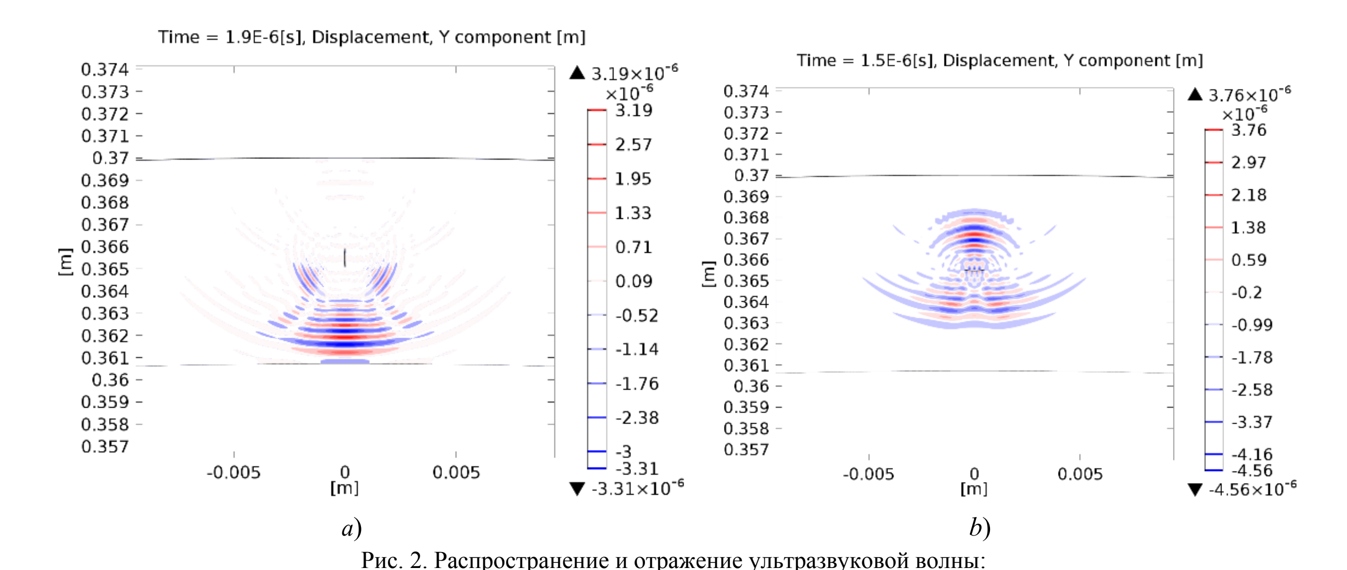


Рис. 2. Распространение и отражение ультразвуковой волны:
трещина расположена вертикально (*a*);
трещина расположена горизонтально (*b*)

Fig.2. Propagation and reflection of ultrasonic wave: crack is located vertically (left); crack is located horizontally (right)

В табл. 1 указано время, которое было затрачено на работу отдельных частей системы, при этом в расчетах было задействовано 25 компьютеров.

Таблица 1
Table 1

Время, затраченное различными компонентами системы

Time spent by different components of the system

Компонент системы **Component**	**Среднее время работы (с)** **Average time (s)**
Решение задачи	1533
Передача данных по сети	1,29
Сохранение данных на диск	0,13
Код исполняющего модуля	0,98

Результаты применения системы и выводы. При применении разработанной системы распределенных вычислений был достигнут существенный прирост производительности при решении задачи конечно-элементного моделирования. Благодаря тому, что исполняющий модуль системы проводит вычисления в многопоточном режиме, аппаратные возможности вычислительных средств были использованы наиболее рационально. Простота организации формата хранения данных и передачи их по сети позволила достигнуть наиболее оптимального использования до-

ступных ресурсов. За счет применения системы резервного копирования данных удалось сократить возможные потери данных, полученных в результате расчетов, при отказе некоторых компонентов системы.

В рамках поставленной задачи эффективность применения разработанной системы распределенных вычислений составила 99.8 %, а общее время проведения расчетов сократилось в 20 раз.

Библиографический список

1. Chih-Fong Tsai, Wei-Chao Lin, Shih-Wen Ke. Big data mining with parallel computing: A comparison of distributed and MapReduce methodologies. Journal of Systems and Software, 2016, vol. 122, pp. 83–92.

2. Gantz, J. The digital universe in 2020: Big data, bigger digital shadows, and biggest growth in the Far East. IDC iView: IDC Analyze the Future, 2012, vol. 2007, pp. 1–16.

3. Shvachko, K., Kuang, H., Sanjay, R., Chansler, R. The Hadoop distributed file system. Mass Storage Systems and Technologies (MSST), 2010 IEEE 26th Symposium on, IEEE, 2010, pp. 1–10.

4. Lakshman, A., Malik, P. Cassandra – A decentralized structured storage system. ACM SIGOPS Oper. Syst. Rev., 2010, vol. 44, no. 2, pp. 35–40.

5. Thusoo, A., Sarma, J.S., Jain, N., Shao, Z., Chakka, P., Anthony, S., Liu, H., Wyckoff, P., Murthy, R. Hive–A warehousing solution over a Map-Reduce framework. Proc. VLDB Endowment, 2009, vol. 2 no. 2, pp. 1626–1629.

6. Foster, I., Kesselman C., Tuecke, S. The Anatomy of the Grid: Enabling Scalable Virtual Organizations. International Journal of Supercomputer Applications and High-Performance Computing, 2001, vol. 15, no 3, pp. 200–222.

7. Frank J. Seinstra, Jason Maassen, Rob V. van Nieuwpoort, Niels Drost, Timo van Kessel, Ben van Werkhoven, Jacopo Urbani, Ceriel Jacobs, Thilo Kielmann, Henri E. Bal. Jungle computing: Distributed supercomputing beyond clusters, grids, and clouds. Grids, Clouds and Virtualization, London: Springer, 2011, pp. 167–197.

8. Таненбаум, Э. Распределенные системы: принципы и парадигмы / Э. Таненбаум. — Санкт-Петербург : Питер, 2003. — 877 с.

9. Soloviev, A.N., Sobol, B.V., Vasiliev, P.V. Ultrasonic Location of Inner Crack Defects in a Compound Elastic Cylinder Using an Artificial Neural-Network Apparatus. Russian Journal of Nondestructive Testing, 2016, vol. 52, no. 3, pp. 119–124.

10. Ватульян, А. О. Обратные задачи в механике деформируемого твёрдого тела / А. О. Ватульян. — Москва : Физматлит, 2007. — 224 с.

11. Willcox, M.A. A Brief Description of NDT Techniques. Toronto: NDT Equipment Limited, 2003. 54 p.

References

1. Chih-Fong Tsai, Wei-Chao Lin, Shih-Wen Ke. Big data mining with parallel computing: A comparison of distributed and MapReduce methodologies. Journal of Systems and Software, 2016, vol. 122, pp. 83–92.

2. Gantz, J. The digital universe in 2020: Big data, bigger digital shadows, and biggest growth in the Far East. IDC iView: IDC Analyze the Future, 2012, vol. 2007, pp. 1–16.

3. Shvachko, K., Kuang, H., Sanjay, R., Chansler, R. The Hadoop distributed file system. Mass Storage Systems and Technologies (MSST), 2010 IEEE 26th Symposium on, IEEE, 2010, pp. 1–10.

4. Lakshman, A., Malik, P. Cassandra – A decentralized structured storage system. ACM SIGOPS Oper. Syst. Rev., 2010, vol. 44, no. 2, pp. 35–40.

5. Thusoo, A., Sarma, J.S., Jain, N., Shao, Z., Chakka, P., Anthony, S., Liu, H., Wyckoff, P., Murthy, R. Hive—A warehousing solution over a Map-Reduce framework. Proc. VLDB Endowment, 2009, vol. 2 no. 2, pp. 1626–1629.

6. Foster, I., Kesselman, C., Tuecke, S. The Anatomy of the Grid: Enabling Scalable Virtual Organizations. International Journal of Supercomputer Applications and High-Performance Computing, 2001, vol. 15, no 3, pp. 200–222.

7. Seinstra, F., Maassen, J., van Nieuwpoort, R.V., Drost, N., van Kessel, T., van Werkhoven, B., Urbani, J., Jacobs, C., Kielmann, T., Bal, H.E. Jungle computing: Distributed supercomputing beyond clusters, grids, and clouds. Grids, Clouds and Virtualization, London: Springer, 2011, pp. 167–197.

8. Tanenbaum, E. Raspredelennye sistemy: printsipy i paradigmy. [Distributed systems: principles and paradigms.] St.Petersburg: Piter, 2003, 877 p. (in Russian).

9. Soloviev, A.N., Sobol, B.V., Vasiliev, P.V. Ultrasonic Location of Inner Crack Defects in a Compound Elastic Cylinder Using an Artificial Neural-Network Apparatus. Russian Journal of Nondestructive Testing, 2016, vol. 52, no. 3, pp. 119–124.

10. Vatulyan, A.O. Obratnye zadachi v mekhanike deformiruemogo tverdogo tela. [Inverse problems in deformable solid mechanics.] Moscow: Fizmatlit, 2007, 224 p. (in Russian).

11. Willcox, M.A. A Brief Description of NDT Techniques. Toronto: NDT Equipment Limited, 2003, 54 p.

Поступила в редакцию 15.09.2017
Сдана в редакцию 15.09.2017
Запланирована в номер 20.10.2017

Received 15.09.2017
Submitted 15.09.2017
Scheduled in the issue 20.10.2017

Об авторах:

Соловьев Аркадий Николаевич,
профессор, заведующий кафедрой «Теоретическая и прикладная механика» Донского государственного технического университета (РФ, 344000 г. Ростов-на-Дону, пл. Гагарина, 1), доктор физико-математических наук, профессор,
ORCID: http://orcid.org/0000-0001-8465-5554
solovievarc@gmail.com

Васильев Павел Владимирович,
старший преподаватель «Информационные технологии» Донского государственного технического университета (РФ, 344000 г. Ростов-на-Дону, пл. Гагарина, 1),
ORCID: http://orcid.org/0000-0003-4112-7449
liftzeigen@mail.ru

Подколзина Любовь Александровна,
программист кафедры «Информационные технологии» Донского государственного технического университета (РФ, 344000 г. Ростов-на-Дону, пл. Гагарина, 1),
ORCID: http://orcid.org/0000-0001-9476-5802
podkolzinalu@gmail.com

Authors:

Soloviev, Arkady N.,
professor, Head of the Department of Theoretical and Applied Mechanics (Strength of Materials), Don State Technical University (RF, 344000, Rostov-on-Don, Gagarin sq., 1), Dr.Sci. (Eng.), professor,
ORCID: http://orcid.org/0000-0001-8465-5554
solovievarc@gmail.com

Vasiliev, Pavel V.,
senior lecturer of the Information Technologies Department, Don State Technical University (RF, 344000, Rostov-on-Don, Gagarin sq., 1),
ORCID: http://orcid.org/0000-0003-4112-7449
liftzeigen@mail.ru

Podkolzina, Lyubov A.,
programmer of the Information Technologies Department, Don State Technical University (RF, 344000, Rostov-on-Don, Gagarin sq., 1),
ORCID: http://orcid.org/0000-0001-9476-5802
podkolzinalu@gmail.com